

Unsupervised learning as a complement to convolutional neural network classification in the analysis of saccadic eye movement in spino-cerebellar ataxia type 2

Catalin Stoean¹, Ruxandra Stoean¹, Roberto Antonio Becerra-García², Rodolfo García-Bermúdez³, Miguel Atencia², Francisco García-Lagos², Luis Velázquez-Pérez⁴, and Gonzalo Joya²

¹ University of Craiova, Romania
{catalin.stoean, ruxandra.stoean}@inf.ucv.ro

² Universidad de Málaga, Spain
idertator@gmail.com, matencia@ctima.uma.es, lagos@dte.uma.es,
gjoya@uma.es

³ Universidad Técnica de Manabí, Ecuador
rodgarberm@gmail.com

⁴ Centro para la Investigación y Rehabilitación de las Ataxias Hereditarias, Cuba
velazq63@gmail.com

Abstract. This paper aims at assessing spino-cerebellar type 2 ataxia by classifying electrooculography records into registers corresponding to healthy, presymptomatic and ill individuals. The primary used technique is the convolutional neural network applied to the time series of eye movements, called saccades. The problem is exceptionally hard, though, because the recorded saccadic movements for presymptomatic cases often do not substantially differ from those of healthy individuals. Precisely this distinction is of the utmost clinical importance, since early intervention on presymptomatic patients can ameliorate symptoms or at least slow their progression. Yet, each register contains a number of saccades that, although not consistent with the current label, have not been considered indicative of another class by the examining physicians. As a consequence, an unsupervised learning mechanism may be more suitable to handle this form of misclassification. Thus, our proposal introduces the k-means approach and the SOM method, as complementary techniques to analyse the time series. The three techniques operating in tandem lead to a well performing solution to this diagnosis problem.

Keywords: Classification; Convolutional Neural Networks; Unsupervised Learning; k-means; Self-Organizing Maps; Saccadic eye movement

1 Introduction

Spino-cerebellar ataxia of type 2 (SCA2) is an incurable neurodegenerative disorder that progressively and, at first imperceptibly, affects the nervous system.

It can be diagnosed by very expensive means such as genetic analysis, and its course is visible in the impairment of certain body movements. But currently, the easiest, cheapest and most widely available procedure is based on electrooculography, by recording and examining the weak electrical potentials generated by the eye movement of a person tracking the trajectory of an object. These movements, induced by an abrupt displacement of the object, are called saccades. They have proved to be a valuable marker in common neurological disorders and their form can be used to perform a diagnosis at a pre-clinical stage of the disease. Consequently, the computational classification of these saccades can not only support a correct distinction between healthy, presymptomatic and ill people, but more importantly an early detection of the presymptomatic cases, thus triggering a timely medical assessment and intervention.

In the last couple of decades, medicine has become the playground of traditional machine learning techniques, such as random forests, support vector machines and neural networks [1,3,5,8,10] and more recently of deep learning methods [4,7,11]. The current problem has been tackled by shallow learning methods, however only to grasp the complexity of the classification task: healthy and presymptomatic registers are easily mistaken for each other [2]. This paper therefore goes further in exploring the potential of saccade classification through deep convolutional neural networks (CNN).

An electrooculographic test consists in a point alternatively appearing at each side of a screen, thus inducing a particular angular deviation in the patient's eyes. The evaluation proceeds by repeating the same object's trajectory several times, and the corresponding saccadic samples form a register, which is labelled by the expert either from its subjective analysis or from additional extra knowledge. The vast majority of saccades in the register of an ill person are clearly distinct from those of a healthy person. However, most saccades in the register of a presymptomatic person are practically indistinguishable from the standard healthy one, whereas a small number present a form slightly (but appreciably) different. Hence, supervised classification of registers comprising both positive and negative examples might get confused in learning the correspondence between a saccade and its label from the whole register. This is the case of CNN, thus our study also appoints two unsupervised approaches to perform a label-free analysis of the saccadic patterns, namely k-means (KM) and self-organizing maps (SOM). Additionally, the two methods provide an informative visualization of the form and disposition of saccades that provides more insight into the problem, unlike black box learning of the CNN. The creation of an ensemble out of the three methods is finally considered, leading to a 93.75% test accuracy on registers.

The paper is organized as follows. Section 2 describes the data and the pre-processing steps before feeding it to the learners. Section 3 outlines the three chosen techniques, with their architecture and supplementary mechanisms for the current problem. The experimental setup and results are presented in Section 4. Section 5 draws the conclusions and suggests directions for future improvement.

2 Data

Electrooculograms are electrophysiologic signals obtained by electrooculography, which measures the electrical potential between the cornea and Bruch's membrane. This potential varies with the angular position of the eye, allowing us to track different eye movements such as saccades. The data used in this work was extracted from electrooculograms recorded with a sampling frequency of 200 Hz.

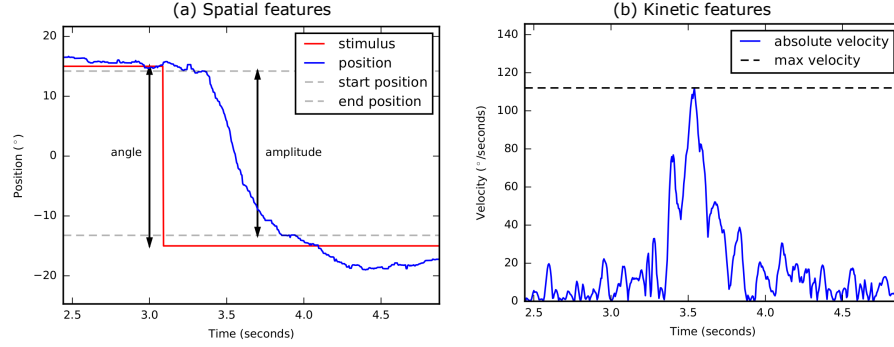


Fig. 1: Useful saccadic spatial and kinetic features

We have registers of 88 subjects at different health status: 29 of them are healthy people (control subjects), 29 are subjects with initial symptoms (presymptomatic individuals), and 30 are already suffering from active SCA2. For each person we have registers corresponding to electrooculograms recorded using protocols with different stimuli angles (10° , 20° , 30° , 60°). After calibration, the electrical potential samples are converted to angular position time series, and the velocity profiles are computed together with some important features, shown in Figure 1. The saccade extraction process is described in the following procedure:

1. Position signals are filtered using a median filter of 9 points.
2. Velocity profiles are calculated using a Lanczos method with 11 points, and the output filtered again with another median filter of 9 points.
3. For detecting the saccades, KM with 2 clusters (having only a preprocessing role here) is applied to samples of the velocity profile. The samples in the cluster with high velocities are considered belonging to a saccade and the other ones to a fixation. Saccadic samples grouped together are selected as individual saccades. The sample with the minimum abscissa value (timestamp) in the group is considered the onset point, and the sample with the maximum abscissa value, the offset point.
4. For each pair of onset and offset points:
 - (a) If the amplitude of the saccade is too low or too high the saccade is considered noisy and discarded. Skip the remaining steps.

- (b) Extract the saccade window using the following procedure:
 - Find the sample index with the maximal velocity between the onset and offset points of the saccade, and define it as mvi .
 - Define the saccade window as the set of 192 samples in the interval $[mvi - 96, mvi + 95]$.
- (c) The central value of the saccade, defined as the average of the minimal and maximal values, is subtracted from saccade windows, thus resulting centred saccades with null central value.
- (d) Left and right saccades can be distinguished where in the former the onset has a lower value than the offset. Right saccades are flipped using a symmetrical transformation with respect to the vertical axis, thus resulting in all the saccades with the same direction.

Finally, 6124 saccade windows of 192 samples each are obtained and vertically stacked resulting in a 6124×192 matrix X , which is then normalized component-wise into the interval $[-0.5, 0.5]$ to enhance the performance of the optimization algorithms involved in the learning process. The normalized matrix X_N results from the following formula: $X_N = \frac{X - \min(X)}{\max(X) - \min(X)} - 0.5$, where the operations \min, \max act over all the matrix components.

The three classes of the problem are denoted as follows: C (control, healthy subjects), P (presymptomatic cases) and S (sick patients). The number of saccades per register in each class ranges in the interval $[49, 169]$ for C, $[38, 172]$ for P, and $[6, 169]$ for S.

3 Methodology

The architectural design of the CNN and the tuning of the KM and SOM for the task at hand are presented in the following subsections.

3.1 Convolutional Neural Networks

Since saccades are time series data, a 1D CNN (with temporal convolutions) is considered. Its architecture is chosen after manual testing. According to extensive preexperimentation, the CNN models applied to this task tend to overfit. Hence several dropout layers are interposed in between the feature extraction and the classification steps. The flow of layers is established as follows:

- A pair of convolutional 1D layers with size 3, a number of filters equal to 128 and a ReLU activation function.
- A max pooling 1D layer with a window size of 2 and a stride of 2.
- A dropout layer with rate 0.4.
- A second pair of convolutional layers with 256 filters of size 3.
- Another dropout layer with rate 0.4.
- A global average pooling 1D layer to further limit overfitting [6].
- A last dropout layer with rate 0.8.

- The final dense layer with the three outputs, corresponding to the given labels. Experiments showed that a sigmoid activation function works better than the usual softmax one, due to the overlapping nature of the outcomes.

The number of epochs is set to 100, the batch size to 25 and the optimizer is Adam. Data are split into training, test, and validation sets by considering whole registers, although the model is trained on individual saccades, whose class is given by the register to which it belongs. The model with the best accuracy on the validation saccadic samples is the selected one.

After training, the saccades in the test set are then taken one by one and labeled with the class predicted by the CNN model. The majority label in one register establishes its class. A register contains several saccades and the medical decision is made on the observed behavior of all these examples. As such, there are several samples that exhibit features distinct from the expected shape for that class. As a consequence, supervised learning will wrongly attribute some samples to classes that correspond to the shape of those series. Therefore, an unsupervised treatment might discover a more accurate grouping of saccades, according to shapes and not to labels given to a whole register. Accordingly, two conceptually different representatives of this type of learning are used for an unsupervised analysis of the problem in the next section.

3.2 K-Means

The simplest form of unsupervised learning is represented by the KM algorithm. The KM approach used herein follows the standard procedure for training: k cluster centroids are established starting from random positions and moved as samples are assigned to them based on proximal Euclidean distances. From pre-experimentation, it is observed that the value for k must be higher than 7, which gives a first confirmation of the multimodality of the saccadic forms present in the data set. Once the different shapes of the given saccades are discovered, the connection between clusters and the three classes of the problem has to be established.

Algorithm 1 outlines the entire KM procedure, from the generation of centroids (line 1) to the labeling of test cases. The determination of the profiles for each of the three categories is performed in the validation phase. Saccades in every validation register are attributed to a generated cluster centroid by Euclidean distance (line 5). Then, for each register, the percentage of saccades that are assigned to each cluster is determined (line 7). At this point, the three labels of the problem are also taken into account and the average percentage is now computed over all registers for each category (line 9). A profile for every label, regarding the percentage of saccades corresponding to each cluster, is therefore obtained. However, these profiles have many intersection points. It is therefore of interest to grasp what is the difference between them, i.e. which clusters are more prominent for each label (those that have more saccades assigned or, on the contrary, less examples as opposed to the amount for the other classes). Consequently, the difference between the cluster values of one category versus

the average of the other two is next calculated by Manhattan distance (line 13). An average of this vector of differences is computed (line 14) and those positions that have values above it denote the clusters that are discriminating for that class (line 15). A ranking of the importance of each of these prominent clusters still has to be quantified and therefore weights proportional to the corresponding value above the average of discerning differences give the measure of this degree (line 16).

In the test phase, the form for each register (in percentage of its saccades attributed to each of the k clusters) is acquired (line 22). Finally, the weighted Euclidean distance (on the base of the weights calculated in line 16) between this form and the profile of each class (obtained in the validation phase—line 9) is calculated (line 23), naturally taking into account only the discriminative clusters (line 15). The distances corresponding to the three classes are then divided by the number of positions found important for each label (line 24). The minimum distance points to the label that will be predicted by the model for the current test register (line 25).

3.3 Self Organizing Maps

Although simple and direct, KM may be a too general algorithm for clustering the temporal saccadic samples. SOM, on the other hand, introduces a more sophisticated dimensionality reduction that is more appropriate for time series analysis, while at the same time provides a different visualization angle of the learning. The SOM generates a two-dimensional map from the initial high-dimensional input, based on the competition between neurons in response to the training saccades and a neighborhood function that preserves the topology of the initial space.

Once the map is unsupervisedly produced from unlabeled saccades in the training phase, the mapping of the test registers proceeds in the following manner. The winning neuron is determined for each saccade in a test register. For that position, each training saccadic sample mapped within the same place is collected and the class to which it belongs gets one vote. The test saccade thus gets a triplet of votes, corresponding to votes for each label of the training samples found in that position. The register is assigned the sum of the triplets of each of its saccades and the class with the maximum value gives its final label. In case of equality, the leftmost (less severe) condition is taken as the label.

4 Experimental Results

The data is split into training-validation-test with the percentages of 40%-40%-20% of randomly taken registers from each category: 12-12-5 registers for C, 12-12-5 for P and 12-12-6 for S.

The following comparative results come from a split where the number of saccades for training is 901 C - 1039 P - 491 S (2431 in total), for validation 977

Algorithm 1 KM detection of saccadic centroids, class profiles and test predictions

Require: Data set of saccadic registers and the three outcomes

Ensure: k centroids and predicted labels for test observations

```

1: Determine the  $k$  centroids from the training saccades
2: for each label do
3:   for each validation register do
4:     for each saccade do
5:       Attribute to a cluster centroid
6:     end for
7:   Compute the percentage of saccades attributed to each cluster
8:   end for
9:   Calculate the average percentage of saccades for every class
10: end for
11: for each label do
12:   Compute the average of the validation profiles of the complementary two labels

13:   Get the vector of differences between the profile of the current label and the
      computed average profile of the other categories
14:   Average the obtained vector of differences
15:   Take positions whose values are above average in the vector as class defining
      clusters
16:   Calculate weights for each position proportional to the distance to the average
17: end for
18: for each test register do
19:   for each saccade do
20:     Attribute to a cluster centroid
21:   end for
22:   Compute the percentage of saccades attributed to each cluster
23:   Compute the weighted distances between the percentage form of the register and
      the validation profile of each label, taking into account only the class defining
      clusters
24:   Divide the distances by the number of positions found for each class
25:   Label the register with the class of the validation profile that led to the minimum
      distance
26: end for

```

C - 845 P - 771 S (2593 in total) and for test 360 C - 421 P - 319 S (1100 in total). There are accordingly 2238 C saccades, 2305 P and 1581 S ones.

Given the random nature for the initial weighting in CNN and SOM, and for the KM initialization of centroids, 30 runs are conducted for each method in order to be conclusive of the obtained test results. The three methods are implemented in Python, using the following libraries: Keras with TensorFlow as back end (CNN), Scikit-learn (KM) and MiniSom (SOM). The architecture and corresponding parametrization of the CNN is appointed after manual selection.

The value for k in KM is also the result of manual tuning. Figure 2 shows in the first plot the 11 centroids discovered by the KM on the training data. Each saccade from the validation and from the test set is then compared to these cluster centers. For every validation register the amount of saccades that are attributed to each cluster (the assignment is set by the minimum Euclidean distance) is computed and a percentage of the saccades distribution is further obtained (line 7 in Algorithm 1). Then, averages are computed over the registers with the same class for validation (line 9): the results are illustrated in the last 3 plots from the second row. The next step regards the identification of the discriminating centroids for every label in turn, based on these computed validation class averages. The discriminative clusters are selected for each class against the other two in such a way that they are either much smaller or much larger than for the complementary labels (line 15). The distinctive centroids for every class in turn are illustrated in the plots 2, 3 and 4 from the first row: dotted line means that the amount of saccades in that register should be smaller for that cluster, while continuous ones indicate the reverse. It should be underlined that the colors for the clusters are kept similar for all plots, be that they contain lines or bars.

The shapes for the C and P classes are very similar, as observed from the profiles for each label found in the validation phase. However, the discriminative clusters are not identical for the two, as indicated in the second plot from the first row: a larger number of saccades closer to the light blue centroid is indicative for the C class, which is one of the reasons why register P020 is wrongly assigned to this class. Conversely, the light blue line is dotted for S (fourth plot in the first row), indicating that a small number of saccades assigned to that centroid is illustrative for this class.

The values for the sigma (of the neighborhood function) and learning rate parameters within SOM are tuned on the validation set by searching through combinations in the set $\{0.9, 1.2, 1.4, 1.6, 1.8\}$ for sigma and $\{0.1, 0.2, 0.3, 0.4, 0.5\}$ for the learning rate. For tuning and for overcoming the stochastic nature of the algorithm, for each setting a number of 5 repeats is considered. The best classification accuracy on the validation set at the register level is achieved for a sigma of 0.9 and a learning rate of 0.1. Another setting that led to the same accuracy result on the validation set was a sigma of 1.2 and a learning rate of 0.3. The number of iterations for training is established at 2000 and the size of the map is chosen to be 40 by 40. Although smaller values for the map size lead to faster running time, the accuracy of the results decreases in these cases,

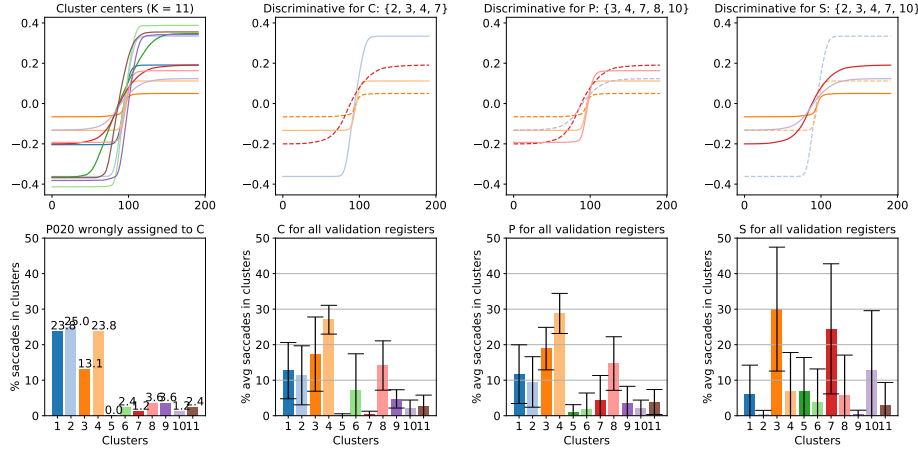


Fig. 2: The plots in the first row indicate the cluster centers as found by the KM in the first picture, followed by the centroids that are discovered as important for each class in turn. The first plot on the second row indicates the profile of the saccades in a presymptomatic register wrongly classified and the subsequent plots show the mean percentages of saccades from the validation set that are assigned to each class in turn. The numbers and colors shown in the plots are consistent among all of them. The centroids in dotted line indicate that the amount of saccades associated with that cluster should be smaller than the corresponding average number from the other two classes; continuously drawn ones mean the opposite.

while larger values for this parameter (it was tried only up to 50) did not lead to significant improvement. Using these settings, the training phase conducts to the map illustrated in Figure 3. Although the problem has only three classes, there are many more clusters obtained by SOM. This comes in line with the KM results which showed that better results were obtained when the number of clusters was high, concretely 11. Moreover, there are many cells in which winning neurons are chosen from different classes—especially the C class is frequently mixed with the P category.

Figure 4 shows the confusion matrices for the three methodologies. The misclassified test registers in the 30 repeated runs for each of the three methods are illustrated in Figure 5. Although for all three matrices the number of confused registers of types C and P are identical, the actual mistaken registers are not the same: as observed in Figure 5, P020 is the only one that is confused by both KM and SOM. Therefore if an ensemble is created out of the three methods by majority vote, there is only one error out of 16 registers, leading to a 93.75% test accuracy.

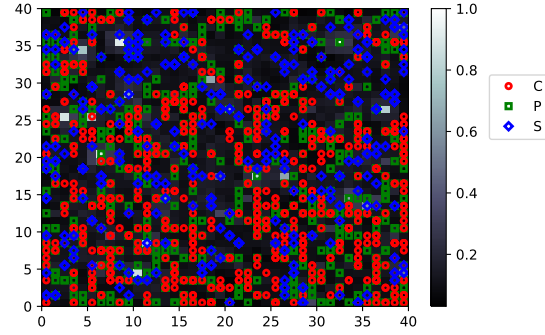


Fig. 3: SOM representation with the winning neurons as obtained from the training phase.

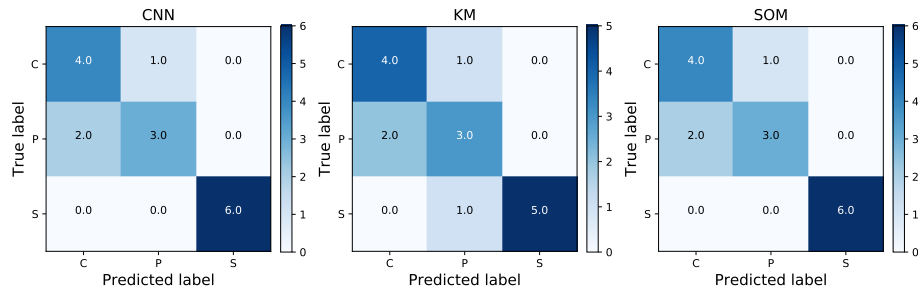


Fig. 4: The confusion matrices for CNN, KM and SOM. The mistaken registers by each model in turn are illustrated in Figure 5.

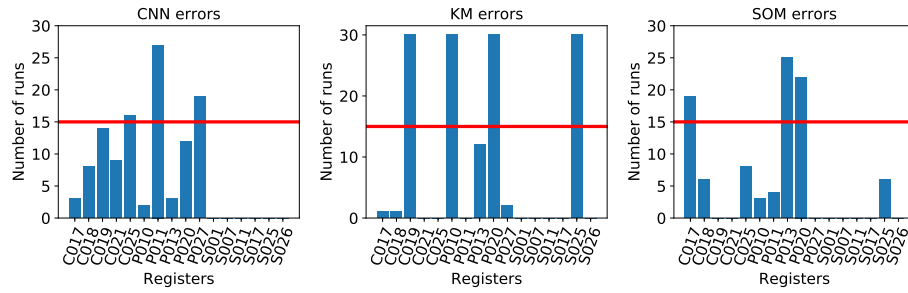


Fig. 5: The number of times the class of each test register is mistaken out of 30 different runs. When a register is mistaken in more than half of the 30 times, it is considered wrongly classified.

5 Conclusions and Future Work

The paper aims to discover the insights of SCA 2 diagnosis from electrooculography data and does so through representatives of both supervised and unsupervised learning. Three techniques are employed for saccade pattern recognition in connection to the three classes (healthy, presymptomatic and ill) of the problem: a convolutional neural network, a k -means and a self-organizing map.

KM performs the worst of the three as a single decision maker (with four wrongly attributed registers from all three categories), which was expected, as it is the simplest form of learning. The CNN and the SOM mistake three registers, two of which are presymptomatic and one healthy. Moreover, it was interesting to see that the three algorithms have each a very different perspective upon the saccades-class correspondence, which demonstrates both the multiple facets of the problem and the efficiency of the ensemble, with only one final wrongly labeled test register and a corresponding accuracy of 93.75%. Although each of the chosen techniques misclassifies more presymptomatic cases than samples from the complementary classes, the overall ensemble mistakes only one, which makes it a suitable solution for the task of an early recognition of the disease.

Future work will target both methodology as well as data enrichment. On the technical side, other classifiers should be included in the ensemble, for example recurrent neural networks that are suitable for capturing temporal behavior. Parametrization of the employed approaches is also an issue related to better performance. Exploring the landscape for the parameters of the CNN (number and type of layers, kernel attributes, initial weighting) and of the unsupervised learners (number of clusters in KM, map size in SOM) can be for instance properly undertaken by evolutionary computation [9].

As far as the data is concerned, the computational support for the diagnosis of SCA 2 has only taken so far into account saccadic movement. There are however some other preclinical indicators of the disease that should be further included as input for the learning methodologies. Evidence that saccadic slowing characterizes the early stages of the condition was brought forth in [12]. Therefore, apart from the movement, the velocity of the saccades should also be investigated computationally. The same study also discovered that the saccadic slowing appears during the presymptomatic stage of the disease only for 60 target amplitude. Therefore, amplitude should be taken into account as a supplementary pattern. Finally, since SCA2 is also a hereditary disease, the genetic markers are expected to bring new discriminating potential within the learning process.

Acknowledgements

This work has been partially supported by the following institutions: the Spanish Ministry of Science, Innovation and Universities, through the Plan Estatal de Investigación Científica y Técnica y de Innovación, Project TIN2017-88728-C2-1-R, and the University of Málaga-Andalucía-Tech through the Plan Propio

de Investigación y Transferencia, Project DIATAX: Integración de nuevas tecnologías para el diagnóstico temprano de las Ataxias Hereditarias.

References

1. Atencia, M.A., García-Garaluz, E., de Arazoza, H., Joya, G.: Estimation of parameters based on artificial neural networks and threshold of HIV/AIDS epidemic system in cuba. *Mathematical and Computer Modelling* 57(11-12), 2971–2983 (2013)
2. Becerra-García, R.A., García-Bermúdez, R., Joya, G., Fernández-Higuera, A., Velázquez-Rodríguez, C., Velázquez-Mariño, M., Cuevas-Beltrán, F., García-Lagos, F., Rodríguez-Labrada, R.: Data mining process for identification of non-spontaneous saccadic movements in clinical electrooculography. *Neurocomputing* 250, 28–36 (2017)
3. Cleophas, T.J., Zwinderman, A.H.: *Machine Learning in Medicine - a Complete Overview*. Springer (2015)
4. Esteva, A., Robicquet, A., Ramsundar, B., Kuleshov, V., DePristo, M., Chou, K., Cui, C., Corrado, G., Thrun, S., Dean, J.: A guide to deep learning in healthcare. *Nature Medicine* 25, 24–29 (2019)
5. Lichtblau, D., Stoean, C.: Cancer diagnosis through a tandem of classifiers for digitized histopathological slides. *PLOS ONE* 14(1), 1–20 (2019)
6. Lin, M., Chen, Q., Yan, S.: Network in network. *CoRR* abs/1312.4400 (2014)
7. Litjens, G., Kooi, T., Bejnordi, B.E., Setio, A.A.A., Ciompi, F., Ghafoorian, M., van der Laak, J.A., van Ginneken, B., Snchez, C.I.: A survey on deep learning in medical image analysis. *Medical Image Analysis* 42, 60 – 88 (2017)
8. Obermeyer, Z., Emanuel, E.J.: Predicting the future big data, machine learning, and clinical medicine. *New England Journal of Medicine* 375(13), 1216–1219 (2016)
9. Preuss, M., Stoean, C., Stoean, R.: Niching foundations: basin identification on fixed-property generated landscapes. In: Krasnogor, N., Lanzi, P.L. (eds.) *13th Annual Conference on Genetic and Evolutionary Computation (GECCO-2011)*. pp. 837–844. ACM (2011)
10. Stoean, C., Stoean, R., Sandita, A., Mesina, C., Gruia, C.L., Ciobanu, D.: How much and where to use manual guidance in the computational detection of contours for histopathological images? *Soft Computing* (2018), <https://doi.org/10.1007/s00500-018-3029-9>
11. Stoean, R.: Analysis on the potential of an EA-surrogate modelling tandem for deep learning parametrization: an example for cancer classification from medical images. *Neural Computing and Applications* (2018), <https://doi.org/10.1007/s00521-018-3709-5>
12. Velázquez-Pérez, L., Seifried, C., Abele, M., Almaguer-Mederos, L., Wirjatijasa, F., Rodríguez-Labrada, R., Santos-Falcón, N., Sánchez-Cruz, G., Tejeda, R., Canales-Ochoa, N., Fetter, M., Ziemann, U., Klockgether, T., Medrano-Montero, J., García Rodríguez, J.C., Laffita Mesa, J., Auburger, G.: Saccade velocity is reduced in presymptomatic spinocerebellar ataxia type 2. *Clinical Neurophysiology* 120, 632–5 (2009)